

基于分层思想的 AUV 路径规划算法研究

徐言民^{1,2)} 昌 政^{1,2)} 赵俊超^{1,2)} 关宏旭^{1,2)} 王 岩³⁾

(武汉理工大学航运学院¹⁾ 武汉 430063) (内河航运技术湖北省重点实验室²⁾ 武汉 430063)
(长江镇江航道处³⁾ 镇江 21200)

摘要:自主水下航行器(autonomous underwater vehicle, AUV)对认识、研究、开发海洋有着重要的意义,其路径规划的研究对 AUV 的航行安全及工作效率提高起着重要的作用. 针对水下复杂环境、障碍物不规则、规划困难等问题,提出适用于三维空间路径规划的空间分层路径规划方法. 通过分层的方式表达空间,简化环境模型,并构建了路径规划的目标函数. 使用逆向逐步搜索路径优化方法,提高了路径实用价值. 仿真结果表明,设计路径规划方法简便有效,适用于复杂水下环境.

关键词:自主水下航行器;复杂环境;路径规划;空间分层;路径优化

中图分类号:U674.941

doi:10.3963/j.issn.2095-3844.2018.06.001

0 引言

新世纪以来很多国家都把海洋列为国家的长远发展计划,为了在海洋的开发中夺得先机,很多都着力于发展更为精密的海洋运载工具. AUV 是一种水下机器人,习惯上称为自主式水下航行器(autonomous underwater vehicle, AUV),AUV 最初主要应用于军事和科研,如水下搜索、监视、侦查、猎雷、作战海洋学、通信、导航、反潜作战等^[1]. 后随着 AUV 技术的逐渐成熟,AUV 在商业领域中也崭露头角,用作海底石油管道的监测、维修,以及钻井平台附近水域溢油监测等任务. 随着科学技术的进一步发展,未来 AUV 将会朝着小型化、混合式、群体化、远程化、智能化等方向发展^[2].

AUV 作为无人航行器的一种,航行在水面以下不同的深度,水中的浅滩、暗礁、岛屿、洋流等都是 AUV 航行过程中的障碍物. 为保证 AUV 的航行安全,必须对其进行航路规划^[3]. 常用路径规划算法有模拟退火法、人工势场法、禁忌搜索算法、栅格法、C 空间法等^[4],但在三维空间的路径规划使用不多. 随着智能仿生算法的发展,如蚁群算法、粒子群算法、遗传算法等在复杂环境的路径

规划中适用性更强,在三维空间路径规划中使用更加广泛. 文献[5]对萤火虫算法进行了改进,提出了一种适用于 AUV 在三维空间中的路径规划方法,相比传统路径搜索算法更加灵活,通过添加辅助规划算子,实现了 AUV 路径的快速规划. 文献[6]运用遗传算法对水中机器鱼的路径规划做了研究. 文献[7]将粒子群算法和遗传算法与蚁群算法相融合,较使用传统的蚁群算法规划三维航迹,提高了航迹规划的速度与效率,增加了算法的高效性. 通过对蚁群算法的改进,提高了算法在三维空间的路径规划效率. 三维空间路径规划存在环境建模困难、规划约束多等难题,使得算法搜索能力欠缺、结构复杂计算量大等问题. 引入分层思想来表达空间环境,降低了环境建模的难度,简洁的算法结构能高效的规划出一条安全路径.

1 环境模型的建立

针对三维空间下的通航环境建模,一般有采用的方法有两种,一种是基于分层的思想,将三维空间投射到一系列的平面上,在二维空间的平面上进行环境建模^[8];另一种方式是采用单元分解的方法,将三维空间划分为单元格,对 AUV 的三

收稿日期:2018-10-07

徐言民(1976—):男,教授,博士,主要研究领域为通航安全、船舶防撞、自动控制

维路径规划中,采用了八叉树模型划分空间^[9-10],朱大奇等^[11-12]在研究 AUV 路径规划问题时,使用了均匀栅格模型,将三维空间分割为一个均匀的栅格地图。

建立空间直角坐标系 $O-XYZ$ 表示整个空间环境,将 XOY 划分为 $n \times n$ 的网格,将海底地形的深度值与 Z 轴对应,因 Z 轴方向从原点向上,将相应地形的水深数据转为高度值,用高度值表示 XOY 平面上每一点地形情况,最后生成相应的环境模型,为简化计算,对所涉及的长度和距离等除以选取的艇长进行量纲一的量化。图 1 为 25×25 的三维环境模型示意图。

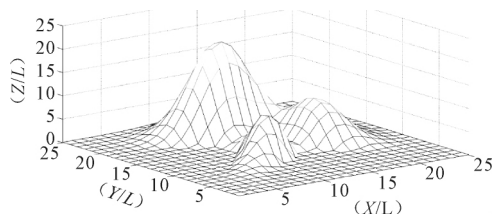


图 1 三维环境示意图

2 路径规划算法设计

2.1 规划路径的表示

根据生成的环境模型,采用分层思想,将空间沿着坐标系 $O-XYZ$ 中的 X 轴,以 dx 等分成若干个 YOZ 平面,见图 2。

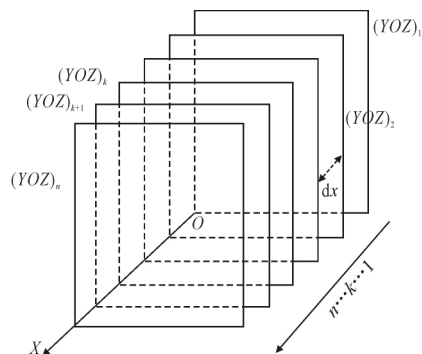


图 2 空间分层示意图

由图 2 可知, X 轴被等分为 n 份,一共可生成 n 个 YOZ 平面, $((YOZ)_1, (YOZ)_2, \dots, (YOZ)_k, \dots, (YOZ)_n)$. 从空间中一点 (x_s, y_s, z_s) 到另一点 (x_e, y_e, z_e) 的路径就可以表示为一系列在 YOZ 平面上的点的连线。

2.2 环境信息的转化

通过将空间沿 X 轴等分成 n 个平面,将空间中的环境信息也分散到每一个 YOZ 平面中,即整个空间环境将被 n 个 YOZ 平面分割,每一个 YOZ 平面都是空间环境的一个切面。根据每一个

断面上点的高度值,可以确定规划路径在每个 YOZ 面上的可行区域,见图 3。图中虚线为环境空间中某一切面 YOZ 。

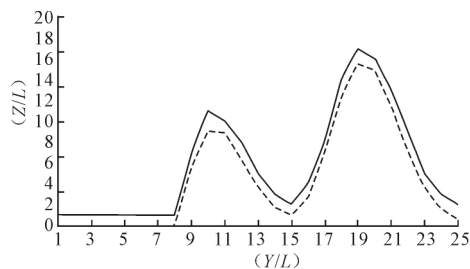


图 3 环境切面示意图

将每个切面再进行网格化处理,分成 $n \times n$ 的网格点,与之对应的建立一个 $n \times n$ 的矩阵用来存储每一个网格点的环境信息。因为高度值并不都是整数,为了便于矩阵存储信息,同时保证路径可行区域的安全性,将高度值进行了取整加一的膨胀处理,图 3 实线表示处理后的空间切面。

图 3 将切面 YOZ 分为 25×25 的网格,同时建立一个 25×25 的 0 矩阵,将高度值不为 0 的路径不可行区域所对应的矩阵元素全部表示为 1,见图 4。0 对应图 4 中“*”,1 对应图 4 中“g”。

1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	0
g	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0	*
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1	0
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	1
g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g	g
1	g	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	0

图 4 切面 YOZ 对应矩阵示意图

2.3 路径评价函数的确定

在路径规划中,总是希望所规划的路径距离障碍物越远越好,路径的长度越短越好,为了找出空间中两点间的最优的路径,就必须要对路径进行评价。考虑到 AUV 的续航能力以及规划航线的安全性,评价函数分别从路径的总长度和路径的危险程度两个方面对规划的路径进行评价。

路径长度的评价函数 规划的路径在切面 $(YOZ)_{i-1}$ 上的点为 p_{i-1} 其坐标为 $(x_{i-1}, y_{i-1}, z_{i-1})$,在切面 $(YOZ)_i$ 路径点在上方的点为 p_i 其坐

标为 (x_i, y_i, z_i) ,则这两个相邻面之间的路径长度可以用式(1)进行表示.

$$L_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \quad (1)$$

在式(1)中由于 $dx = x_i - x_{i-1}$ 是一个定值,连接每个切面中的路径点就可以组成一条路径,当AUV沿着选取的路径航行时,AUV的路径长度评价价值 F_1 计算模型见式(2).

$$F_1 = \int_0^{L_i} \sum_{i=1}^{n-1} ((dx)^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2)^{\frac{1}{2}} dl \quad (2)$$

危险程度的评价函数 在切面上的每个路径点,在一定步长 $r(r \in Z)$ 内,都会存在 R 个网格点,假设这 R 个网格点中有 $m(m < 8)$ 个点为不可行点,即代表切面矩阵中为1的元素,则此时该路径点的危险就可以记为式(3),当选取的路径点周围全部为不可行点,即 $D_i = 1$ 时,此时表示选取的路径点不可行,需要重新选取路径点.

$$D_i = \frac{m}{R} \quad R = r^2 - 1 \quad (3)$$

把规划路径中所有路径点的危险度进行相加就可以得到整条路径的危险程度评价价值 F_2 计算模型见式(4).

$$F_2 = \int_0^{D_i} \sum_{i=2}^{n-1} D_i dd \quad (4)$$

路径的总评价函数 路径规划的目的是使路径长度与危险程度都达到最优值,对于建立的评价函数,就相当于对两个目标进行优化求解.最直观简洁的方法就是采用加权的方法,通过对两个不同的目标设置不同的权重,然后再相加,从而把两个不同目标的优化求解转化成一个目标的优化求解,最后得到路径规划总评价函数模型,见式(5).

$$\begin{aligned} \min F &= KF_1 + (1-K)F_2 = \\ &\left\{ K \left(\int_0^{L_i} \sum_{i=1}^{n-1} ((dx)^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2)^{\frac{1}{2}} dl + \right. \right. \\ &\quad \left. \left. (1-K) \int_0^{D_i} \sum_{i=2}^{n-1} \left(\frac{m}{R} \right) dd \right\} \quad (5) \end{aligned}$$

F 为路径总评价价值,权重系数 K 取值范围 $[0, 1]$,根据不同的环境情况,危险度和路径长度评价函数权重不同.

2.4 算法的实现

设置所需规划路径的起点坐标为 $S(x_s, y_s,$

$z_s)$,终点坐标为 $E(x_e, y_e, z_e)$.根据两点之间线段最短的原理,首先连接路径的起点与终点得到一条向量 SE ,给出路径的搜索方向,且向量 SE 与每一个切面都会有一个交点记为 $(p_1^0, p_2^0, p_3^0, \dots, p_n^0)$.在进行路径点搜索的时候,首先从 p_1^0 开始检测 p_2^0 是否在 $(YOZ)_2$ 平面的可行区域中,若在则平面 $(YOZ)_1$ 与平面 $(YOZ)_2$ 之间的路径就可以用向量 $p_1^0 p_2^0$ 来表示,若不在,则下一个路径点需要根据下述方法来选择.向量 SE 与 $(YOZ)_i$ 平面的交点 p_i^0 不在平面 $(YOZ)_i$ 的可行区域内,设路径在 $(YOZ)_{i-1}$ 平面上的点为 p_{i-1} ,把 p_{i-1} 投影到下一个平面 $(YOZ)_i$ 上为 p'_{i-1} ,然后以 p'_{i-1} 为基准点搜索 $(YOZ)_i$ 面中的可行点,设置路径搜索过程中每一次搜索的步长为一个网格单元长度.当搜索的路径点距离障碍物的边缘小于两个单元长度时就是不可行点,记为1,反之当路径点与障碍物的距离大于两个网格单元长度时就是可行点,记为0,并把上述0和1作为矩阵存储起来,得到整个 $(YOZ)_i$ 平面可行点的集合见式(6).

$$\begin{aligned} &((x_i, y_1, z_1), (x_i, y_1, z_2), \dots, \\ &(x_i, y_i, z_i), \dots, (x_i, y_a, z_b)) \quad (6) \end{aligned}$$

式中: a, b 为 $(YOZ)_i$ 平面中横纵坐标的编号,且 $(a \leq n, b \leq n)$; n 为 $(YOZ)_i$ 平面中被分割的最大网格数量.

设 $p_i(x_i, y_i, z_i)$ 为 $(YOZ)_i$ 平面中的一个可行点,首先根据式(7)计算点与点之间的距离 L_j ,然后根据式(8)计算点与点 p_i^0 之间的距离 L_k .

$$L_j = \sqrt{(y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \quad (7)$$

$$L_k = \sqrt{(y_i - y_i^0)^2 + (z_i - z_i^0)^2} \quad (8)$$

则任意可行点与点及点 p_i^0 之间的距离之和就可以用 $L_i = L_j + L_k$ 表示,由于可行点的数量有很多所以 L_i 的值不是唯一的,只有当 L_i 取最小时才能够被选为下一个路径点.

L_i 取最小时, L_j 和 L_k 的取值都最小. L_j 取最小值表示所选取的路径点与上一路径点之间的距离最短,符合路径规划中整体路径最短的要求, L_k 取最小值表示平面 $(YOZ)_{i-1}$ 与平面 $(YOZ)_i$ 之间的路径向量 $p_{i-1} p_i$ 在总体方向上与向量 SE 比较接近,这也符合路径规划中路程最短的要求,具体方法见图5.此时平面 $(YOZ)_{i-1}$ 与平面 $(YOZ)_i$ 之间的路径长度就为向量 $p_{i-1} p_i$ 的绝对值 $|p_{i-1} p_i|$,即

$$\begin{aligned} |p_{i-1} p_i| &= \\ &\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \quad (9) \end{aligned}$$

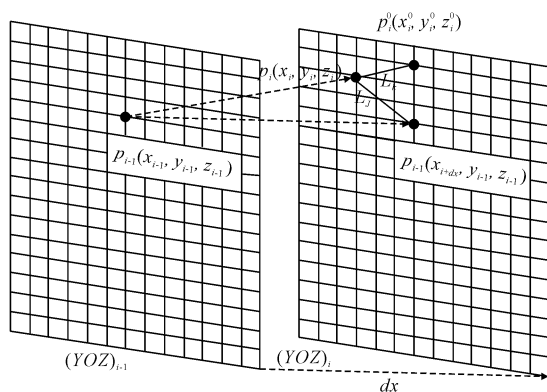


图5 路径点选取方法

平面 $(YOZ)_i$ 中的路径点搜索出来后, 连接点与终点 E , 得到一条新的向量 $p_i E$, 此时向量 $p_i E$ 与平面 $(YOZ)_i$ 到平面 $(YOZ)_n$ 会重新产生一系列的点, 把向量 $p_i E$ 作为选取下一个路径点时的参考向量, 重复上述选取路径点的方法, 依次进行就可以得到 $(YOZ)_i$ 平面与 $(YOZ)_{i+1}$ 平面之间最短的安全路径, 见图 6. 把每一段最短的安全路径依次连接起来就可以构建一条从起点到终点的安全路径.

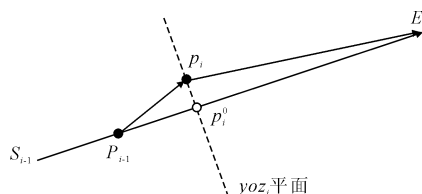


图6 路径选取示意图

3 路径规划

本文将路径规划的起终点设定在了环境障碍物两侧, 起点坐标设置为 $[1, 8, 6]$, 终点坐标设置为 $[25, 23, 10]$, 权值 $K=0.5$. 使用 MATLAB 软件对模型进行求解, 为了验证所设计算法的效果, 使用仿生算法中蚁群算法与本设计算法进行比较, 见图 7.

由图 7 可知, 本算法与蚁群算法均能完成路径的规划. 从规划路径的长度进行比较, 本算法的规划路径长度较蚁群算法的短; 从规划路径中航向变化的改变幅度来看, 本算法的规划路径中航向变化幅度较蚁群算法小; 整体看来, 本文所设计算法规划结果较蚁群算法要好, 但对于规划路径中航向改变的频繁程度, 两种算法都需要有所提高.

4 规划路径的优化

对应本文所建立的规划模型, 规划航路为 D

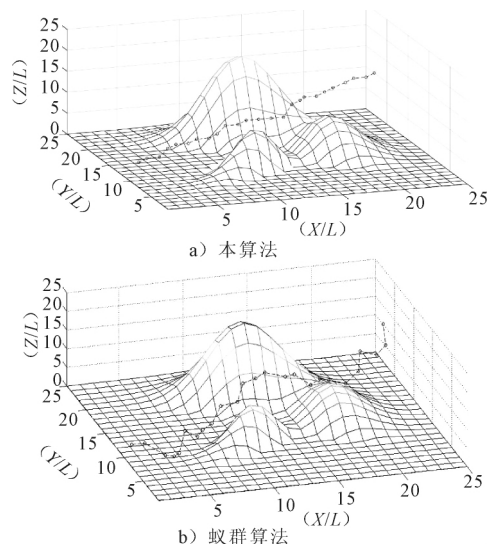


图7 路径规划结果示意图

维向量加起始点的一组向量. 为了保证规划路径的精度, 在规划路径时设置的维数较多, 规划结果呈现出曲线或者较多段线段的形态, 这也导致了规划的路径中航向变化频繁的问题, 实际中并不适用. 为了更贴近实际, 仍需要对规划路径进行优化.

所求路径可表示为 $P = (p_1, p_2, \dots, p_n)$, 包括起终点, 路径一共要穿过 n 个 YOZ 平面. 以终点为基点, 跳过相邻路径点, 按反向顺序选取路径点进行检验, 判断与构成线段所穿过路径点与间的所有 YOZ 平面是否均为可行区域, 即线段与平面交点均位于相应平面的可行区域内, 则从路径 P 中剔除与相邻的上一个路径点, 选取下一个路径点 p_{n-3} 进行检验; 否则保留上一个路径点, 再重新以为基点, 重复上述操作. 直至检验判断遍历至起点为止, 就可得到一条优化后的路径 P' , 见图 8.

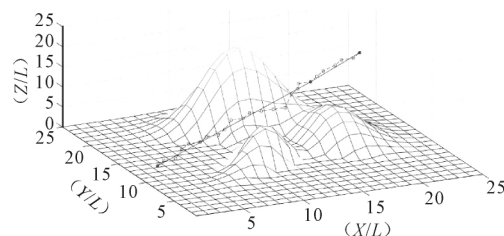


图8 优化后路径示意图

5 结束语

引入分层的思想重新对环境进行表达, 提出了一种适用于复杂水下三维环境的 AUV 路径规划方法. 通过仿真实验, 验证了所提出的路径规划方法的可行性, 且较于常规的蚁群算法, 此算法结构更简单高效, 仿真结果更好, 提出的路径优化方法有效减少了规划路径中的转向点, 解决了频繁

转向的问题,使规划结果更具实用性.

参考文献

- [1] 李喜刚,蔡远利. 基于改进蚁群算法的无人机路径规划[J]. 飞行力学, 2017, 35(1): 52-56.
- [2] 李连鹏,苏中,解迎刚,等. 基于遗传算法的机器鱼水中路径规划[J]. 兵工自动化, 2015(12): 93-96.
- [3] 朱颢东,孙振,吴迪. 基于改进蚁群算法的w移动机器人三维路径规划[J]. 华中师范大学学报:自然科学版, 2016, 50(6): 849-855.
- [4] ZHANG D, XIAN Y, LI J, et al. UAV path planning based on chaos ant colony algorithm[C]. IEEE, 2015: 81-85.
- [5] 乔慧芬. 机器人路径规划算法研究[D]. 太原: 中北大学, 2015.
- [6] 肖玉杰,邱志明,石章松. UUV国内外研究现状及若干关键问题综述[J]. 电光与控制, 2014(2): 46-49.
- [7] FIORELLI E, LEONARD N E, BHATTA P, et al. Multi-AUV control and adaptive sampling in Monterey bay[J]. IEEE Journal of Oceanic Engineering, 2006, 31(4): 935-948.
- [8] 曹晓霖. 水下无人航行器及其路径规划技术的发展现状与趋势[J]. 电子世界, 2014(13): 118-119.
- [9] 赵丹,戚龙. 基于蚁群-粒子群融合算法的无人机三维航迹规划研究[J]. 吉林化工学院学报, 2017, 34(3): 93-98.
- [10] 王磊. 海洋环境下水下机器人快速路径规划研究[D]. 哈尔滨: 哈尔滨工程大学, 2007.
- [11] 朱大奇,孙兵,李利. 基于生物启发模型的AUV三维自主路径规划与安全避障算法[J]. 控制与决策, 2015, 30(5): 798-806.
- [12] 潘昕,吴旭升,侯新国,等. 基于遗传蚂蚁混合算法的AUV全局路径规划[J]. 华中科技大学学报(自然科学版), 2017, 45(5): 45-49.

Study on AUV Path Planning Algorithm Based on Spatial Stratification

XU Yanmin^{1,2)} CHANG Zheng^{1,2)} ZHAO Junchao^{1,2)} GUAN Hongxu^{1,2)} WANG Yan³⁾

(Wuhan University of Technology, School of Navigation, Wuhan 430063, China)¹⁾

(Hubei Province Key Laboratory of Inland Waterway Transport
Technology, Wuhan 430063, China)²⁾

(Changjiang Zhenjiang Waterway Agency, Zhenjiang 21200, China)³⁾

Abstract: Autonomous Underwater Vehicle (AUV) is of great significance to the understanding, research, and development of the ocean. The study of AUV path planning plays an important role in improving AUV's navigation safety and working efficiency. In order to solve the problems of complicated underwater environment, irregular obstacles and difficult planning, a spatial stratification path planning method suitable for three-dimensional spatial path planning was proposed. The space was expressed in a hierarchical way, the environmental model was simplified, and the objective function of path planning was constructed. Finally, the reverse step-by-step search path optimization method was used to improve the practical value of the path. Simulation results show that the design path planning method is simple and effective, and it is suitable for complex underwater environments.

Key words: autonomous underwater vehicle; complex environment; path planning; spatial stratification; path optimization